Slackware Live

Operating principle

Prerequisites

The following Slackware packages are required:

- **"a/dosfstools"**,
- **"a/gptfdisk"**,
- **"a/grub"**,
- **"a/mkinitrd"**,
- **"ap/squashfs-tools"**.

Quick start

The following commands build a live system from the running Slackware system:

```
build-slackware-live.sh --init / /tmp/live
    #builds the live system initRD & copies the kernel
build-slackware-live.sh --module / /tmp/live 0-slackware-live
    #creates a squashfs module of the system (and mounted
medias...)
build-slackware-live.sh --iso /tmp/live /tmp/slackware-live.iso
    #generates an ISO of the live CD (if needed)
build-slackware-live.sh --usb /tmp/live /dev/sdx
    #copies the live system to USB (if needed)
```

Example using a clean (non running) system

- Setup a clean system from the running one:

  ```
  installpkg -root /mnt/system /mnt/cdrom/slackware/a/*.t?z #...
  chroot /mnt/system
     pkgtool
     useradd -m -g users -s /bin/bash liveuser #(if needed)
     usermod -G cdrom,netdev,plugdev,scanner,audio,video,power
  liveuser
     passwd liveuser
     exit #chroot
  ```

- Build the live system:

  ```
  build-slackware-live.sh --init /mnt/system /tmp/live
  build-slackware-live.sh --module /mnt/system /tmp/live 0-
  slackware-live -xz
  build-slackware-live.sh --iso /tmp/live /tmp/slackware-
  live.iso
  ```

System setup

Live system is just an ordinary Slackware

Install the live system like any other Slackware Linux (or derived distribution):

- make a regular install from the distribution install media, then, access this system by mounting its partition from the build system:
  ```
  mkdir /mnt/system
  mount /dev/installation_device /mnt/system
  ```
- or install packages into a subdirectory of the build system (the live system root directory) — with **"installpkg -root /mnt/system ..."** for example —.

Sysprep

Running Sysprep is recommanded before building the live system and is required if you want to speed up live system boot (see **"fastboot"** boot parameter); it consists of the following actions:

- run commands like **"mkfontdir"** (... — ordinary run by **"pkgtool"** during setup);
- run commands like **"fc-cache"**, **"update-pango-querymodules"** (... — run by stock Slackware **"rc.M"** startup script);
- run **"ldconfig"** and **"depmod"**;
- merge **"/etc/passwd"** and **"/etc/group"** system files (if the system is divided into multiple directories);
- set up root and users profile (from **"/etc/skel"**).

- usage: **"build-slackware-live.sh --sysprep dir_1(rw) dir_2(ro) dir_3(ro)..."**
  only the first directory parameter is mandatory; if the system is divided into multiple directories (one per module), list all of them to recompose the whole system; setup changes will be written into the first one
- example: **"build-slackware-live.sh --sysprep /mnt/system-gui /mnt/system-core"**

Live system build commands

*Live media detection relies on filesystem label (defaulting to "**SlackLive**"), hard-coded in "**init**" and "**build-slackware-live.sh**" scripts.*

Setup kernel and initrd

- usage: **"build-slackware-live.sh --init root_dir live_dir [modules_list]"**
  *the "**modules_list**" defaults to "**squashfs:overlay:loop:xhci-pci:ehci-pci:usb-storage:ext4:isofs:usbhid**" (needed for Slackware stock generic kernel — dependencies are automatically added)*
- example: **"build-slackware-live.sh --init /mnt/system /tmp/live"**

Create a SquashFS module for the system

- usage: **"build-slackware-live.sh --module root_dir live_dir module_file [-xz|-gzip] [-optional]"**
  *all directories but "**/sys**", "**/proc**", "**/dev**" and "**/tmp**" are included*
  *"**-xz**" is smaller, "**-gzip**" (defaults) is faster*
- example - for live system build (typically):
  "build-slackware-live.sh --module /mnt/system /tmp/live 0-slackware-live"
- example - to store changes made on live USB while running:
  "build-slackware-live.sh --module /live/changes /live/media 1-changes"

Using multiple modules

The system can be divided into multiple modules (example: "core", "gui", "tools",...) that are loaded in alphabetical order; if a file is present into several modules, the one taken from the last loaded module is used; naming example:

- 1-an_application
- 1-an_other_application
- ...
- 2-gui
- 3-core
- 4-2012-01-01-updates
- 4-2012-02-01-updates
- ...

System on USB device: manual partitioning (discouraged)

For USB booting (live or installed system), the recommended partition scheme is an hybrid / with the following partitions (use **"r"** then **"h"** gdisk commands to add partitions 1 and 2 to MBR):

- a GPT / EFI partition is needed to enable booting; if present, it is the only partition usable by Ms. Windows so far;
- a Linux partition (for the live system); *if the filesystem is not an "ext" one, its label must be set manually; non filesystems are strongly discouraged (no persistent changes nor home directory support)*;
- a boot partition is required by GRUB on GPT disk / key for legacy BIOS (aka) booting.

Number	Start (sector)	End (sector)	Size	Code	Filesystem	Content
1	2048		at least 16 MiB	ef00	fat32	EFI System & misc files
2			at least 1 GiB (depending on live system size)	8300	ext4	Live system
any	34	2047	1007.0 KiB	ef02		GRUB for CSM mode

Copy live system on USB device

*Warning: if the destination is a whole disk or key ("**/dev/sdb**" for example), it is **automaticaly repartitionned** (see above) (and **all data on it are wiped**).*

- usage: **"build-slackware-live.sh --usb live_dir device"**
- example - after initialization and module creation: **"build-slackware-live.sh --usb /tmp/live /dev/sdx2"**
- example - from a running live system: **"build-slackware-live.sh --usb /live/media /dev/sdx"**

Create a live CD/DVD ISO from live system

- usage: **"build-slackware-live.sh --iso live_dir iso_file_name"**
- example - after initialization and module creation: **"build-slackware-live.sh --iso /tmp/live /tmp/slackware-live.iso"**
- example - from a running live system: **"build-slackware-live.sh --iso /live/media /mnt/sdy1/slackware-live.iso"**

Hybrid ISO / USB (discouraged)

*This is given for reference only: copying the live system on the USB as explained above (cf "**--usb**") is a better solution (it supports persistent changes and home directory).*

- Convert ISO: **"isohybrid /path/to/iso"**
- Copy ISO on key (warning, **wipes everything** on key): **"dd if=/path/to/iso of=/dev/sdx"**

Boot parameters

System language and keymap layout

- **"locale"**: system language; example: **"locale=fr_FR.UTF-8"**
- **"keymap"**: keymap layout; example: **"keymap=fr"**; the first two caracters are used for Xorg keymap layout
- **"tz"**: timezone; example: **"tz=Europe/Paris"** (value must be a valid path from **"/usr/share/zoneinfo"**)
- **"hwc"**: hardware clock: **"UTC"** or **"localtime"**

Modules loading

- **"include=module1:module2:..."**: to load selected modules from **"/boot/optional"** directory (module names are the SquashFS file names)
- **"exclude=module1:module2:..."**: to specify the main modules (from **"/boot/modules"** directory) not to load; example: **"exclude=1-gui"**

Persistent home directory storage or system changes

- **"home=/path/to/directory|NFS_resource||label"**: activates persistent home directory;

- **"changes=/path/to/directory|NFS_resource||label"**: activates persistent system changes .

Misc

- **"runlevel=[1-5]"**: overrides default startup runlevel (cf **"/etc/inittab"**)
  on runlevel 5, the user with uid 1000 will be automatically logged in and X session started (if installed); if this user doesn't exists, root user is used instead
- **"copy2ram=yes"**: enables running the live system from memory
- **"useswap=yes"**: enables swap on detected swap partitions
- **"rootpw=*password*"**: defines a root password
- **"fastboot=yes"**: skip **"ldconfig"**, **"depmod"**, **"fc-cache"**, **"update-mime-database"**, **"gtk-update-icon-cache"**, **"update-gtk-immodules"**, **"update-gtk-immodules"** and **"update-pango-querymodules"**

Live system installation

The live system can be installed into a hard disk partition; the result is the same as a clean installation.

Install live system

- usage: **"build-slackware-live.sh --install root_dir device [-auto]"**
  *the "**-auto**" option enables GRUB installation*
- example - from a running live system (typically):
  "build-slackware-live.sh --install /live/system /dev/sdx2 -auto"

Manual boot loader installation - BIOS version

Install GRUB in the MBR with the following commands (given **"/dev/sdx2"** is the Linux installation partition):

```
mkdir /mnt/install
mount /dev/sdx2 /mnt/install
grub-install --boot-directory /mnt/install/boot --target i386-
pc /dev/sdx
#GRUB configuration (see below)
```

If the disk is GPT partitionned, A BIOS boot partition is needed; example:

Number	Start (sector)	End (sector)	Size	Code	Name
1	2048	206847	100.0 MiB	ef00	EFI system partition
2	206848	105064447	50.0 GiB	8300	Linux filesystem
128	34	2047	1007.0 KiB	ef02	BIOS boot partition

Manual boot loader installation - UEFI version

Install GRUB with the following commands (given **"/dev/sdx1"** is the EFI partition and **"/dev/sdx2"** is the Linux installation partition):

```
mkdir /mnt/install
mount /dev/sdx2 /mnt/install
mkdir -p /boot/efi
mount /dev/sdx1 /boot/efi
grub-install --boot-directory /mnt/install/boot --efi-directory
/boot/efi
umount /boot/efi
#GRUB configuration (see below)
```

Manual GRUB configuration

```
mount --bind /dev /mnt/install/dev
mount --bind /proc /mnt/install/proc
mount --bind /sys /mnt/install/sys
chroot /mnt/install grub-mkconfig -o /boot/grub/grub.cfg
umount /mnt/install/dev
umount /mnt/install/proc
umount /mnt/install/sys
umount /mnt/install
rmdir /mnt/install
```

Live system usage

Firefox cache and persistent home directory problem

When using Firefox with persistent home directory feature enabled on an USB flash disk, disk cache should be disabled to avoid continuous writings. This can be done by pointing your Web browser to "**about:config**" (or by editing "**.mozilla/firefox/*PROFILE_NAME*/prefs.js**") and set "**browser.cache.disk.enable**" property to "**false**".

Live Root Over NFS setup

The live system can be booted over the network. Both SysLinux and GRUB2 are supported for PXE booting.

Prerequisites

*UnionFS-FUSE is required since overlay filesystem doesn't support NFS; it is automaticaly included (if available) in the live initrd by the "**build-slackware-live.sh --init**" command.*

- DHCP server ("**n/dhcp**")
- TFTP server ("**n/inetd**", "**n/tftp-hpa**")
- NFS server ("**n/rpcbind**" + "**n/nfs-utils**" + "**n/libtirpc**")
- Patched GRUB2 (see attached patch and grub package)

DHCP setup

- Create a "**/etc/dhcpd.conf**" file with a content like:

```
ddns-update-style none;
subnet 192.168.1.0 netmask 255.255.255.0 {
        range 192.168.1.1 192.168.1.100;
        option routers 192.168.1.254;
        option domain-name-servers 192.168.1.254;
        filename "/boot/grub/i386-pc/core.0";
       next-server 192.168.1.253;
}
```

- Start the server ("**dhcpd**")

TFTP setup

- Populate the "**/tftpboot**" directory with the following commands:

```
grub-mknetdir --net-directory=/tftpboot
cp /path/to/generic-kernel /tftpboot/boot/vmlinuz
cp /path/to/slackware-live/initrd.gz /tftpboot/boot/
```

- Add a "**/tftpboot/boot/grub/grub.cfg**" file (ip string has the following format: "**ip:pxe_server:gateway:netmask**"):

```
menuentry "Slackware-Live" {
        linux /boot/vmlinuz nfsroot=192.168.1.253:/live/media
max_loop=255 locale=fr_FR.UTF-8 keymap=fr tz=Europe/Paris
hwc=localtime
ip=$net_pxe_ip:192.168.1.253:192.168.1.254:255.255.255.0
        initrd /boot/initrd.gz
}
```

- Enable tftp (edit "**/etc/inetd.conf**") and start inetd service

NFS setup

- In the "**/etc/exports**" file, add a line like:

```
/live/media
192.168.1.0/255.255.255.0(ro,no_root_squash,async,no_subtree_check)
```

- Start the nfs service (or reload configuration with "**exportfs -r**")

- grub-tftp.patch.zip (2014-09-18)